2018

Automated Cleaning of Identity Label Noise in A Large-scale Face Dataset Using A Face Image Quality Control

Mohamad Al jazaery
moaljazaery@mix.wvu.edu

Automated Cleaning of Identity Label Noise in A Large-scale Face Dataset Using A Face Image Quality Control

Mohamad Al jazaery

Thesis submitted to the
Benjamin M. Statler College of Engineering and Mineral Resources
at West Virginia University
in partial fulfillment of the requirements
for the degree of

Master of Science
in
Computer Science

Guodong Guo, Ph.D., Chair
Donald Adjeroh, Ph.D.
Xin Li, Ph.D.

Lane Department of Computer Science and Electrical Engineering

Morgantown, West Virginia
2018

Keywords: Convolutional Neural Network, Machine Learning, Deep Learning, Face Recognition, Face Quality, Identity Label Cleaning, Face Dataset

Abstract

Automated Cleaning of Identity Label Noise in A Large-scale Face Dataset Using A Face Image Quality Control

Mohamad Al jazaery

For face recognition, some very large-scale datasets are publicly available in recent years which are usually collected from the internet using search engines, and thus have many faces with wrong identity labels (outliers). Additionally, the face images in these datasets have different qualities. Since the low quality face images are hard to identify, current automated identity label cleaning methods are not able to detect the identity label error in the low quality faces. Therefore, we propose a novel approach for cleaning the identity label error more low quality faces. Our face identity labels cleaned by our method can train better models for low quality face recognition. The problem of low quality face recognition is very common in the real-life scenarios, where face images are usually captured by surveillance cameras in unconstrained conditions.

Our proposed method starts by defining a clean subset for each identity consists of top high-quality face images and top search ranked faces that has the identity label. We call this set the "identity reference set". After that, a "quality adaptive similarity threshold" is applied to decide on whether a face image from the original identity set is similar to the identity reference set (inlier) or not. The quality adaptive similarity threshold means using adaptive threshold values for faces based on their quality scores. Because the inlier low quality faces have less facial information and are likely to achieve less similarity score to the identity reference than the high-quality inlier faces, using less strict threshold to classify low quality faces saves them from being falsely classified as outlier.

In our low-to-high-quality face verification experiments, the deep model trained on our cleaning results of MS-Celeb-1M.v1 outperforms the same model trained using MS-Celeb-1M.v1 cleaned by the semantic bootstrapping method. We also apply our identity label cleaning method on a subset of the CACD face dataset, our quality based cleaning can deliver a higher precision and recall than a previous method.

Acknowledgements

I want to take this opportunity to thank my advisor and committee chair Dr. Guodong Guo for his valuable guidance and relentless support. He inspired me greatly in working through this project with his innovative ideas. I am grateful to him for believing in me and providing me an opportunity to work in his research group and making my Masters program a memorable experience. I also thank the Lane Department of Computer Science and Electrical Engineering for providing me resources and financial support through research assignments. Also, I would like to thank my family and friends for their love and support. Last but not the least, I give thanks to God for giving me strength through difficult times.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem and Motivation

Due to recent advances in using the deep learning techniques for face recognition, the need for large face datasets with accurate identity labels has increased dramatically. To build large datasets, researchers typically collects a large amount of face images from the Internet. But this kind of datasets usually contain identity labels ambiguity. Also, the fact of being large-scale makes them almost impossible to be cleaned from identity (id) label error by just taking a manual approach. Furthermore, these large face datasets are not only filled with outliers with false id labels, but also have different levels of quality. Low-quality face images with low resolutions in addition to uncontrolled poses and illumination conditions are hard to identify. In the current automated id label cleaning methods, low-quality faces are usually removed when trying to handle the id label errors. Developing an automated id label cleaning method which keeps more inlier low-quality face images helps in training better face models to perform low-quality face recognition. The problem of low-quality face matching and recognition happens very often in the real life, where the face images are usually captured by surveillance cameras in unconstrained conditions are compared with passport style high-quality face images.

Different factors can affect the face image quality such as:

- Brightness
- Focus
- Contrast
- Illumination
- Illumination symmetry
- Sharpness
- Compression Quality
- Face Symmetry
- Face Pose

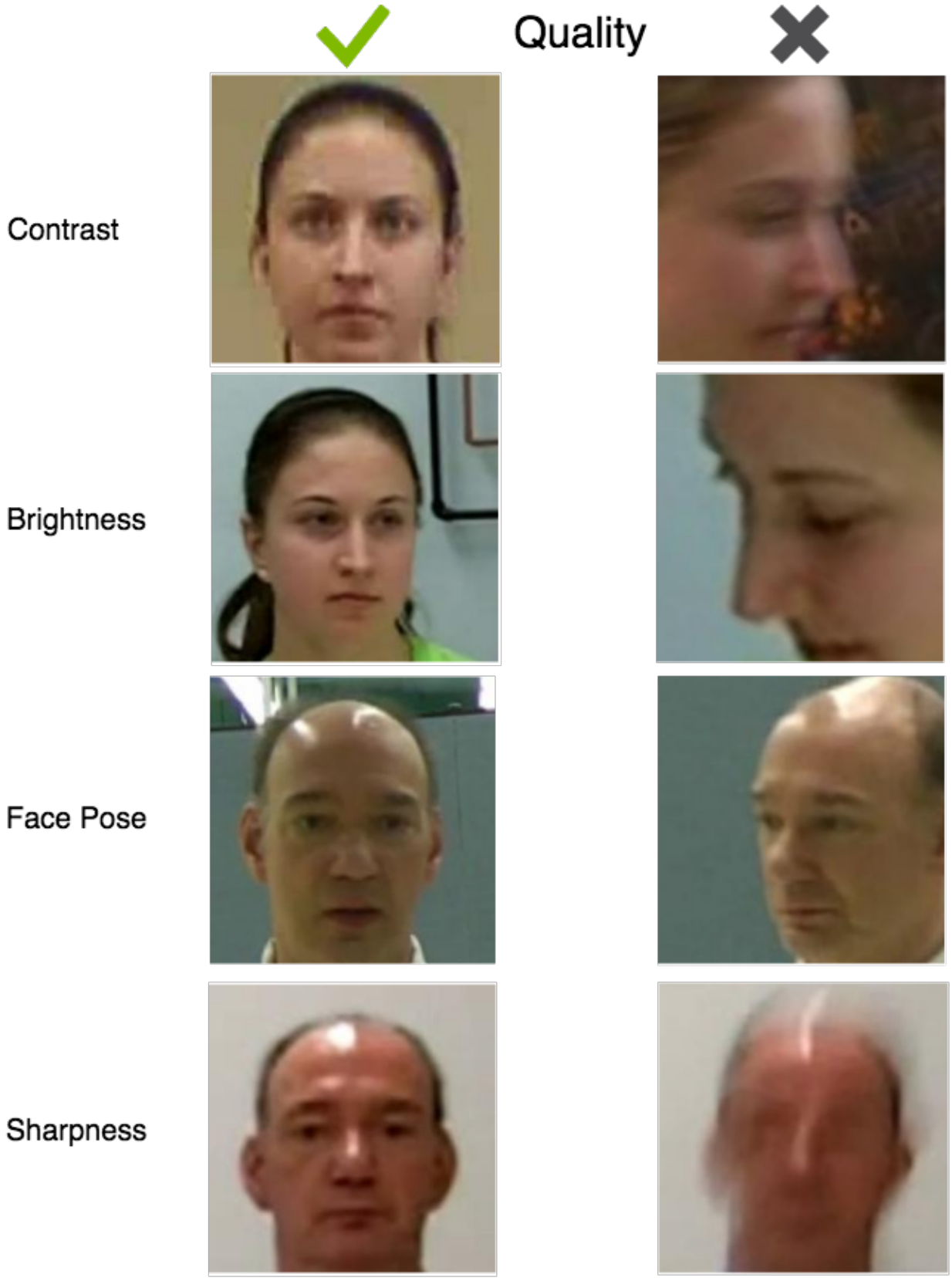


Figure 1.1: Low and high-quality face image examples.

- Face Size

Figure 1.1 shows for some low and high-quality face images examples.

1.2 Thesis Contributions

In this work, we propose a novel approach identity id label errors in a large face dataset with giving a special consideration for the low-quality faces. Our contributions include:

- Developing a novel method to identity label error in a large face dataset using a face image quality assessment which can preserves low-quality inlier faces, face images with correct id labels, while remove the outliers.
- An evaluation of the proposed approach and comparisons to other representative approaches in different aspects, showing that our method can produce better result for low-to-high quality deep face matching than the semantic bootstrapping cleaning approach [2]. Also, a comparison with human annotations shows that our id label cleaning approach achieves higher recall and precision on a larger face dataset than Ng and Winkler's [3].

1.3 Thesis Outline

The remaining of the paper is organized as follows: Chapter 2 presents a prior face dataset cleaning works and approaches. In the second section of this chapter, the face quality assessment (FQA) method is described as a background of our method. In Chapter 3, our novel id label cleaning of a large face dataset using face image quality assessment framework is presented in details with all the steps. In Chapter 4, we compared our method indirectly to semantic bootstrapping [2], by conducting two different low-to-high quality deep face matching experiments. Finally, we compared our id label cleaning output to human annotations as a direct way to evaluate our method. Lastly, concluding remarks and future directions are offered in Chapter 5.

Chapter 2

Background and Literature Review

2.1 Related Face Dataset ID Label id label cleaning Works

Recently a number of large face datasets consisting of unconstrained face images have been constructed. During their construction a varied number of methods were applied to ensure correct annotation and noise removal. Ng and Winkler [3] proposed to identify the outliers by formulating the problem as a quadratic programming (QP) problem that combines the outputs of an outlier detection classifier and a gender classifier, enforcing visual similarity among the outliers and inliers, while at the same time constrains to at most one face per image to be an inlier. Their results on FaceScrub database show that the method can effectively clean the raw data. To clean their VGG-Face database, Parkhi et al. [4] first used human annotators to select the identities having over 90% pure images, then removed erroneous faces in each set automatically using a linear SVM classifier. After that, removed near duplicates by clustering the VLAD descriptor of the images. And lastly, again used human annotators after ranking images within each identity set by decreasing likelihood of being an inlier. Zhang et al. [5] proposed an approach to automatically collect and label large-scale celebrity faces from the web. They named the database Celebrities on the Web (CFW). Using their image annotation system, they analyzed surrounding text of the near-duplicates of each image and were able to provide a set of names corresponding to the celebrities appearing in the image. Using the annotation results and a proposed multimodal name assignment algorithm, they assigned names to faces in the image. They found that the overall error rate of the labels of CFW dataset is 13.93% and a significant portion of CFW dataset (constituting over half of the CFW dataset) achieves an error rate as low as 4.07%. Bansal et al. [6] introduced a new dataset called UMDFaces which has 367,920 face annotations of 8,501 identities. They used a face detection model with a low threshold on the detection score to get a high recall. After that, they used votes from human annotators and calculated a score using a weighted vote based scheme. Finally, by thresholding the score they cleaned the images which were passed by the face detection model. Yi et al. [7] built a large-scale face dataset which includes about 10,000 identities and 500,000 images, called CASIA-Webface. They crawled the IMDb, a well structured website containing rich information of celebrities, to collect the images. Then all images are processed by a multi-view face detector. After that, they used a tag-similarity clustering method to clean the dataset. Later on, to illustrate the quality of CASIA-Webface, they trained a deep CNN

Dataset	Type	identities	Images	Cleaning
FaceScrub	Public	695	141,130	automated
VGG-Face	Public	2,622	$\sim$ 2.6M	hybrid
CFW	Public	421,436	2.45M	automated
UMDFaces	Public	8,501	367,920	manual
CelebFaces	Public	5,436	87,628	unknown
YTF	Public	1,595	3,425	hybrid
WebFace	Public	10,575	494,414	automated
MS-Celeb-1M	Public	100K	$\sim$ 10M	automated
MS-1M-2R	Public	79,077	5,049,824	automated
Facebook	Private	4K	4.4M	unknown
Google	Private	8M	100-200M	unknown

Table 2.1: Large-Scale Face Datasets. In the cleaning method column, automated means there was no human involvement in the cleaning process. Hybrid method means it used a combination of automated and human processing.

on it. Sun et al. [8] created a dataset called Celebrity Faces dataset (CelebFaces) by first collecting the celebrity names that do not exist in LFW [9], then searching for the face images for each name on the web. It contains $87,628$ face images of $5,436$ celebrities from the web and was assembled by searching for the face images for each name on the web. Wolf et al. [10] created the 'Youtube Faces' (YTF) set by using the $5,749$ names of identities included in the LFW data set [9] to search YouTube for videos of these same individuals. They downloaded the top six results for each query and minimized the number of duplicate videos by considering title of two videos with an edit distance less than 3 to be duplicates. Downloaded videos are then split to frames at 24 fps and then detected faces in these videos. Finally, the videos were manually verified to ensure that they are correctly labeled, and no identical videos are included in the database. Wu et al.

[2] used semantic bootstrapping to clean the identity label noise in the MS-Celeb-1M.v1. First, they trained a lightCNN model on the original noisy labeled dataset. Secondly, the trained model is utilized to predict the identity labels of the noisy training dataset. Finally, using a threshold they decided whether accept or reject the prediction according to a condi-

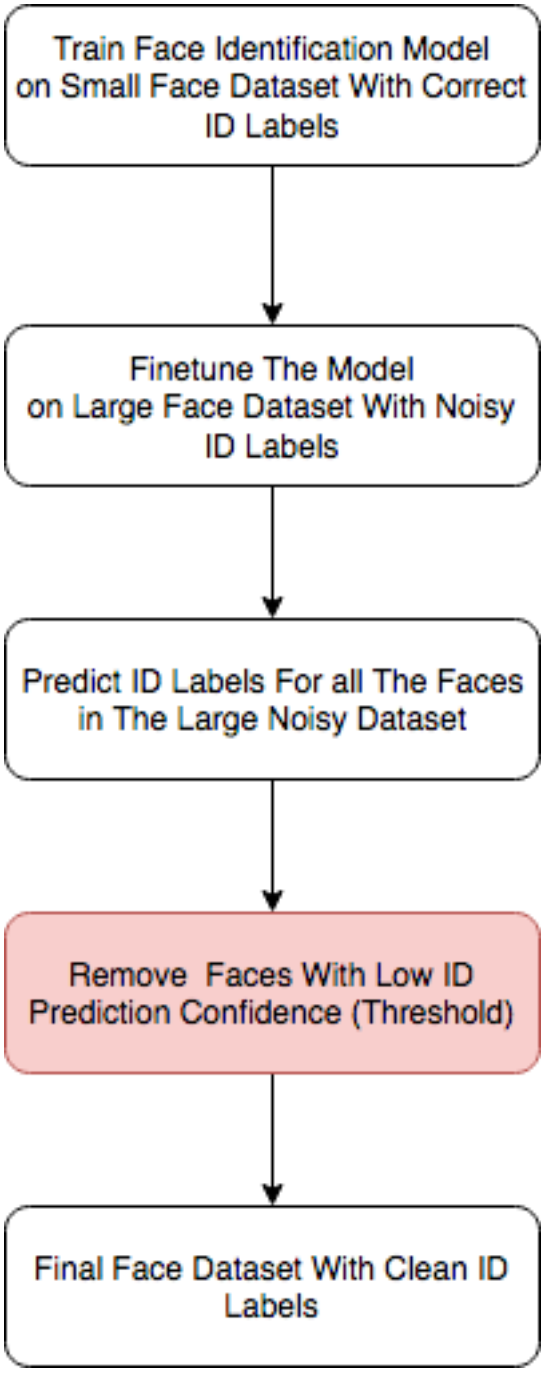


Figure 2.1: The flow chart of identity labels cleaning using semantic bootstrapping method.

tional probability. Fig 2.1 is an illustration of the semantic bootstrapping cleaning method, showing the steps and components. They called their cleaned result set as MS-1M-2R. Table 2.1 gives a comparative view of the different large-scale face databases.

2.2 Face Quality Assessment (FQA)

In general, it is a very difficult problem to explicitly define and quantify the biometric quality of a face image. So far, there have been mainly two categories taken to solve this problem. The first category is to use heuristic features of the face such as resolution of the face region, pose angle, illumination, focus, or expression, to quantify the quality of the face image [11, 12, 13, 14, 15]. The other one is to use a reference image as the template for high-quality and measure some distance metric to find the discrepancy from that reference to the query face image to measure its quality [16, 17]. But these methods do not consider the face recognition method that will be used to match faces. For this reason, these traditional approaches can be considered inflexible and limited in applicability.

Another face image quality assessment method is proposed by Chen et al. [18], which considers face image quality measure in a relative manner. A rank learning based quality assessment approach is used to handle the learning problem. An illustration of this method is shown in Figure 2.2. Furthermore, let us assume that a face identification method is applied on two different datasets D_1 and D_2. Also, assume all images in each of these databases have been taken by following some environmental conditions E_1, and E_2, respectively. If the recognition method F has higher accuracy in D_1 than in D_2, then we denote it as $F(D_1) \succ F(D_2)$. Let, I_p and I_q are two face images such that $I_p \in D_1$ and $I_q \in D_2$, and, $f(\cdot)$ is the function that extracts the feature vector from an image. Then the quality assessment function $Q(\cdot)$ can be defined as

$$Q(I) = \omega^T f(I) \tag{2.1}$$

So, the learning method should optimize ω such that it satisfies the constraints in Eqns. (1), (2) and (3):

$$\omega^T f(I_p) > \omega^T f(I_q), \forall I_p \in D_1, \forall I_q \in D_2 \tag{2.2}$$

$$\omega^T f(I_p) = \omega^T f(I_q), \forall I_p \in D_1, \forall I_q \in D_1 \tag{2.3}$$

$$\omega^T f(I_p) = \omega^T f(I_q), \forall I_p \in D_2, \forall I_q \in D_2 \tag{2.4}$$

The above formulated problem can be converted to a convex max-margin formulation as shown in Eqn. (4) below.

$$\begin{aligned} \text{minimize}(\omega^T{}_2^2 + \lambda_1 \sum \xi_{pq}^2 + \lambda_2 \sum \eta_{pq}^2 + \lambda_3 \sum \gamma_{pq}^2) \\ \text{s.t. } \omega^T(f(I_p) - f(I_q)) \geq 1 - \xi_{pq}, \forall I_p \in D_1, \forall I_q \in D_2 \\ \omega^T(f(I_p) - f(I_q)) \leq \eta_{pq}, \forall I_p \in D_1, \forall I_q \in D_1 \\ \omega^T(f(I_p) - f(I_q)) \leq \gamma_{pq}, \forall I_p \in D_2, \forall I_q \in D_2 \\ \xi_{pq} \geq 0, \eta_{pq} \geq 0, \gamma_{pq} \geq 0 \end{aligned} \tag{2.5}$$

Multiple features have been used to train the model, and a two-level learning method is applied for feature fusion. Let us assume, n types of features are extracted from an image I; then the quality assessment function for the i_{th} feature will be $Q_i(I) = \omega_i^T f_i(I)$; where, $i = 1, 2, \ldots, n$. In learning the first level features, all rank weights ω_i are trained by solving Eqn. (4) for the n different feature functions. Suppose, the vector $\vec{V} = [Q_1(I), Q_2(I), \ldots Q_n(I)]^T$ consists of quality scores of I for the n different features. Then, the second level quality assessment function for I can be defined as $Q_2(I) = \omega_2 \zeta(\vec{V})$. Where $\zeta(\cdot)$ is a n-degree polynomial kernel mapping function. We use $n = 5$ different features and a second order polynomial kernel.

$$\begin{aligned} \zeta(\vec{V}) = [c, \sqrt{2c}Q_1, Q_1^2, \sqrt{2c}Q_2, \sqrt{2}Q_1Q_2, Q_2^2, \sqrt{2c}Q_3, \\ \sqrt{2}Q_1Q_3, \sqrt{2}Q_2Q_3, Q_3^2, \sqrt{2c}Q_4, \sqrt{2}Q_1Q_4, \\ \sqrt{2}Q2Q4, \sqrt{2}Q_3Q_4, Q_4^2, \sqrt{2c}Q_5, \sqrt{2}Q_1Q_5, \\ \sqrt{2}Q_2Q_5, \sqrt{2}Q_3Q_5, \sqrt{2}Q_4Q_5, Q_5^2]^T \end{aligned} \tag{2.6}$$

The second level training gives the values of ω_2 and $Q_2(I)$. Later, $Q_2(I)$ is normalized to the interval $[0, 100]$, rounded to the nearest integer and used as the quality score of face image I.

At first level, RankSVM [18] is trained using five different face recognition features, namely, HoG, Gabor, Gist, LBP and CNN. Then the predicted ranks are used to create second level features using the mapping function. This new feature is used to train RankSVM at second level, which produces the desired quality scores. Figure 2.2 shows the two-level

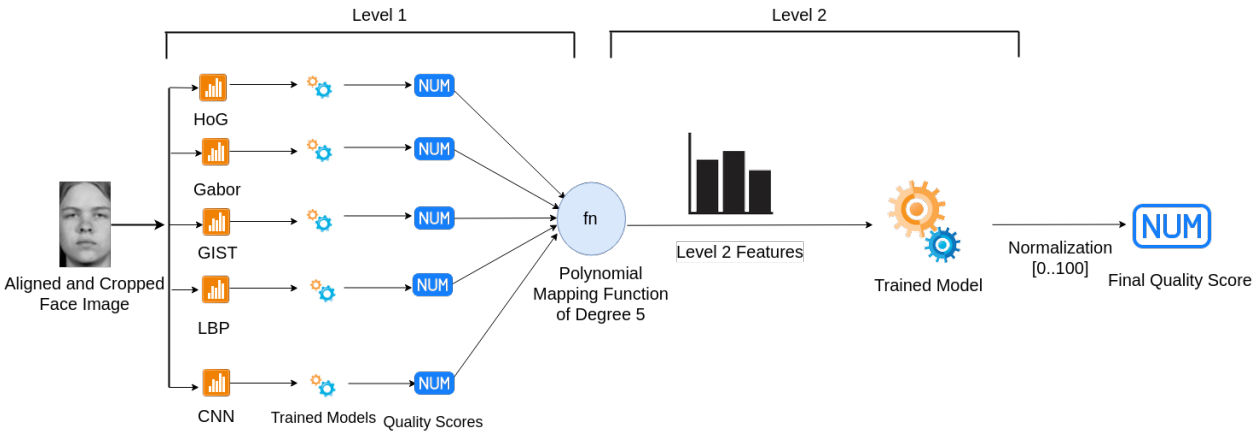


Figure 2.2: Two level learning method to calculate a face quality score.

learning process. A dataset containing controlled, real world and non-face images is used for training. This method can generate scores that agrees with human intuition about face image quality.

2.3 Application of Face Quality in video Face verification

Identity recognition in videos is very challenging recognition task. One identity video usually contains many faces of different qualities. Therefor, using face quality control to choose best quality faces in an identity video could improve the identity feature representations. Consequently, this could improve the video-to-video face identity verification.

We conduct three video-to-video face verification experiments. In each experiment, using N top quality faces are compared to using all the faces in the video. Two different video datasets are used. The First irst dataset is the Video Database of Moving Faces and People (VDMFP) which was collected at the University of Texas at Dallas, in hallways with an unconstrained pose and illumination. The dataset contains same identities performing two different actions: walking and conversation. The other dataset is PaSC dataset was acquired at the University of Notre Dame. All identities performed the same action (out of seven total actions). A handheld and control videos were acquired at the same time for each subject. More information about the datasets and the experiments protocols is available in [19].

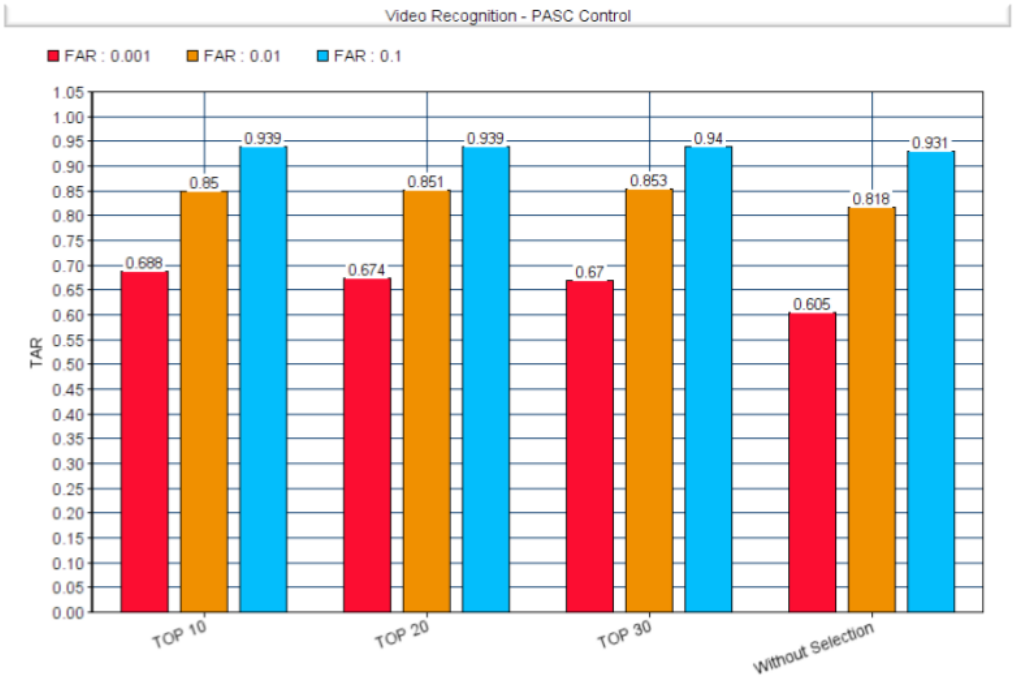


Figure 2.3: Face verification results on PaSC control video dataset using different number of quality-based face selections.

The results show that selecting top quality faces in instead of using all the faces in a video always improves the face verification rate. Figures 2.3, 2.4, 2.5 show the results of our three experiments.

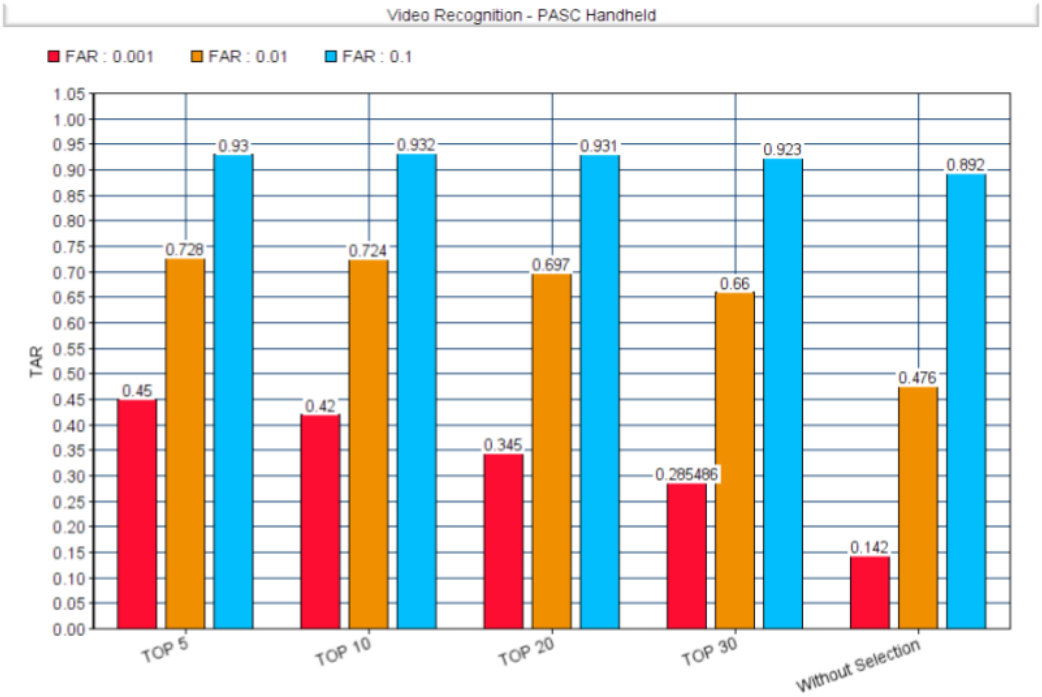


Figure 2.4: Face verification results on PaSC hand-held video dataset using different number of quality-based face selections.

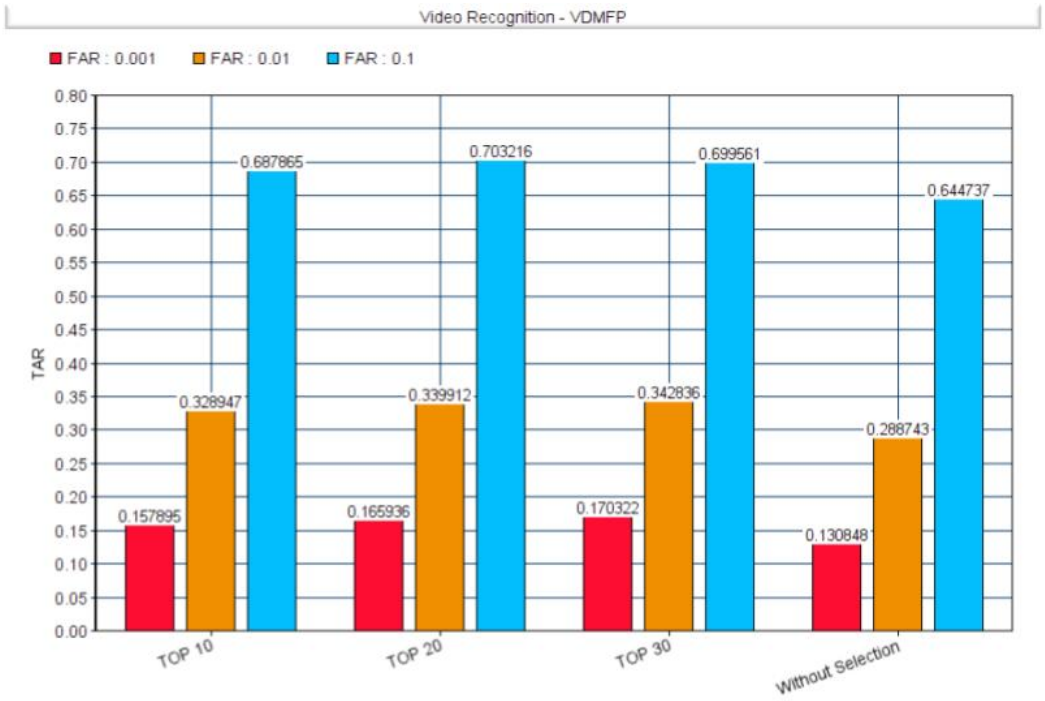


Figure 2.5: Face verification results on VDMFP video dataset using different number of quality-based face selections.

Chapter 3

Methodology

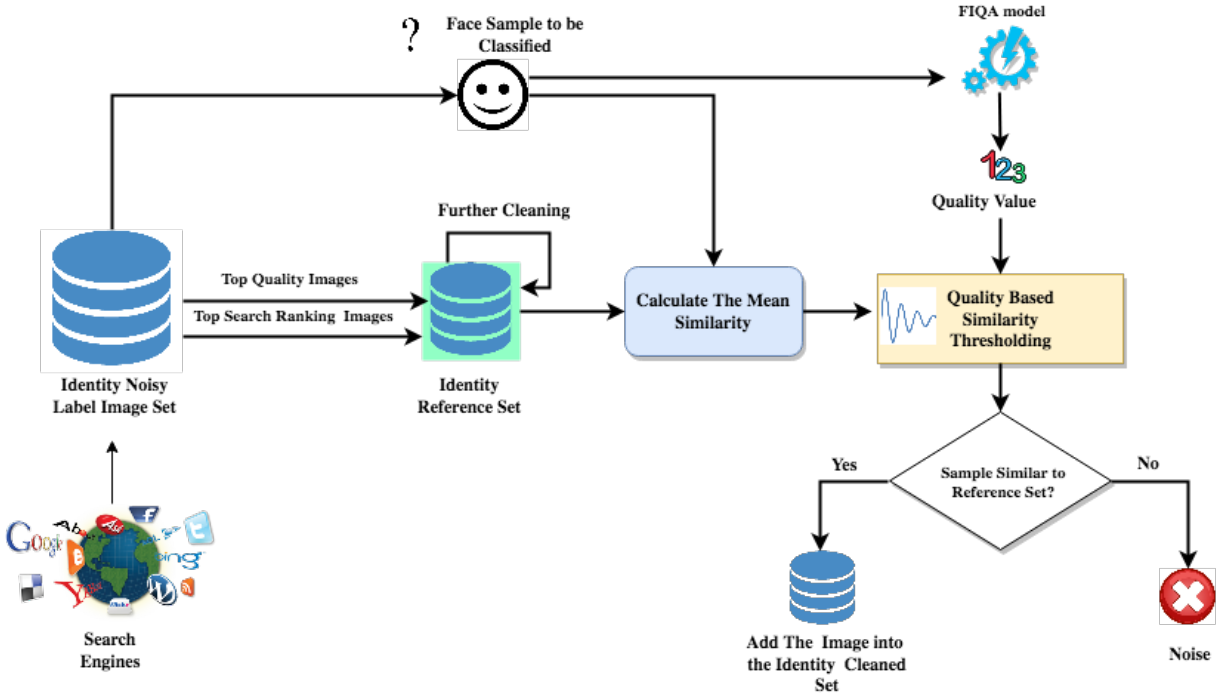


Figure 3.1: The flow chart of our proposed method for cleaning identity label noise using the face quality assessment (FQA). First, the method defines an identity reference faces set out of the noisy identity faces set. Then using quality-based similarity threshold, decide on whether a face from the noisy identity set is similar to the identity reference set. If not, the system considers the face as a noise otherwise the face will be added to the output identity cleaned set.

First, let us define the input and the output of our id label cleaning problem. The input is an identity (id) noisy set of all faces that has the same identity (id) label. Here, noisy means the set includes some outlier faces which do not belong to the assigned identity. The output is a clean identity set by excluding the outliers. The method starts with defining a clean subset using some preliminary assumptions. We call this set the "reference set". Then further id label cleaning is done on this identity reference set. After that, a "quality adaptive similarity threshold" is applied to decide on whether a sample face is similar to the identity reference set (inlier) or not. The quality adaptive similarity threshold means using adaptive threshold values for the faces based on their qualities. Because the low-quality inlier faces have less facial information and are likely to achieve less similarity score to the identity reference than the high-quality inlier faces, using quality adaptive similarity threshold to

classify a sample face may save many low-quality faces from being falsely classified as noise. Figure 3.1 is an illustration of our framework, showing the major steps and components.

We can divide our id label cleaning method into four steps: 1) constructing an initial identity reference set based on some preliminary measures, 2) tuning this identity reference set to become cleaned, 3) applying the quality adaptive similarity threshold, and 4) building the final cleaned set. In the following , we provide explanation of our quality-based id label cleaning method in details. See Algorithm 1 for a procedural description of the method.

3.1 Defining an initial identity reference set

Building an identity reference set which includes face images that have a high probability to be inliers helps in classifying any sample from the original noisy identity set S, such that, samples which are similar to the faces in the identity reference set are considered inliers. Thus, we build an initial identity reference set R_{init} consisting of face images that have a high probability to be inliers. First, because the faces are collected via a search-engine, there is usually a ranking of faces based on their order in the search result, and the faces with high search rankings have a relatively high probability of being inlier faces of an identity. Based on that, the top three search ranked faces related to the identity are added to R_{init}. Secondly, we assume that the majority of the high-quality faces are potential inliers. Based on that, all the face images above the mean quality value $\overline{Q}$ are considered high-quality and added to R_{init}. The mean quality $\overline{Q}$ is the average of the quality scores for all the faces in the face dataset. At this point, R_{init} contains the top high-quality faces and the top three search ranked faces from S.

3.2 Finalizing the identity reference set

To avoid any noise in R_{init}, we estimated a similarity threshold T_{init}, such that, any face in R_{init} that achieves less than threshold T_{init} to the remaining set of faces in R_{init} is not added to the final identity reference set. By excluding these outliers, we create the final identity reference R from R_{init}. In other words, the identity reference set R is a subset of R_{init} where the face image $I \in R_{\text{init}}$ is considered part of R only if its average similarity to the remaining faces in the set R_{init} is above a similarity threshold T_{init}. The identity

reference set R is defined as following:

$$R = \{I \mid \text{sim}(I, R_{\text{init}}) > T_{\text{init}}, I \in R_{\text{init}}\} \tag{3.1}$$

where $\text{sim}(\cdot)$ is the function that calculates the mean similarity between the face features I and average face features in the set R_{init} using cosine similarity measure, T_{init} is an estimated similarity threshold.

3.3 Quality adaptive similarity (QAS) threshold

The quality adaptive similarity (QAS) depends on the face image quality assessment to determine the best similarity threshold for the sample. The method proposed by Chen et al. [18] is used to estimate the quality of each face image. It is a learning to rank based FQA method which uses the ranking SVMs trained on a rank-ordered set of faces. At first, the rank SVMs learn rank weights for five different face image features (HoG, Gabor, Gist, LBP, and CNN features), then the features are fused into a single feature set using polynomial kernel mapping and another weight vector is learned for the fusion feature. To get the predicted score for a face image I, the 5 face image features are extracted and multiplied by their corresponding weight vectors, then fused into a second level feature, and finally multiplied with fusion feature weight. The quality score is then normalized within the range of $0 - 100$. If $f(\cdot)$ is the function that extracts the feature vector from a face image, the quality assessment function $Q(\cdot)$ can be defined as:

$$\boldsymbol{Q(I)} = \text{P}(\boldsymbol{\omega^T f(I)})\boldsymbol{\omega'} \tag{3.2}$$

Where I is a face image, ω is the learned weight vector for first level features, $\text{P}(\cdot)$ is the polynomial kernel mapping function and ω' is the learned weight for the fused features.

As mentioned earlier, to decide whether a sample is an inlier or not, it should be compared to the identity reference set R. To do that a similarity threshold is needed. Since R has mostly high-quality face images, it is highly possible that the low-quality inlier faces achieves less similarity scores than the high-quality ones when comparing them to the identity reference set R. If we try to classify the samples using a strict high similarity threshold, we could falsely classify the inlier low-quality faces as noise. On the other hand, if we use

Algorithm 1: Quality-Based Face Identity label Cleaning

Input : S The identity noisy faces set.

Output: C The identity cleaned faces set.

begin

    // Building the initial identity reference set.

    $R_{init} \longleftarrow \{top.search.ranked.faces(S)\}$

    for $I \in S$ **do**

        /* Add the high-quality face images only. */

        if $Q(I) > \overline{Q}$ **then**

            $R_{init} \longleftarrow R_{init} \cup \{I\}$

        end

    end

    // Building the final identity reference set.

    for $I \in R_{init}$ **do**

        /* Exclude possible outliers. */

        if $mean.sim(I, R_{init}) > T_{init}$ **then**

            $R \longleftarrow R \cup \{I\}$

        end

    end

    // Building the final clean set.

    for $I \in S$ **do**

        /* Classify using the Quality Adaptive Similarity Threshold. */

        if $mean.sim(I, R) > T_{QAS}(I)$ **then**

            $C \longleftarrow C \cup \{I\}$

        end

    end

end

a low threshold, it could lead to many outliers (noise) to be falsely included. To solve this dichotomy, an adaptive similarity threshold is performed, where the threshold goes lower when the face image quality is lower. However, the relation between the quality and the similarity threshold is not strictly linear because the threshold is highly affected by the quality of the face image in the low-quality range whereas the range of middle to high-quality face images has less influence on the similarity threshold. Based on these facts, we define our QAS threshold function $T_{\text{QAS}}(\cdot)$ as following:

$$T_{\text{QAS}}(I) = T_{\text{max}} - \frac{(T_{\text{max}} - T_{\text{min}})}{e^{(Q(I)/2\overline{Q})}} \tag{3.3}$$

where I is the face image, $T_{\text{max}}, T_{\text{min}}$ are the maximum and minimum similarity thresholds, $\overline{Q}$ is the average quality, $Q(I)$ is the function provided in Equation (3.2). T_{QAS} threshold increases faster in the low to mid quality range but slower and smoother for the range above the average quality ($\overline{Q}$). Figure 4.3 shows one example of how the function $T_{\text{QAS}}(\cdot)$ values change for different quality values.

3.4 Building the final clean set

After calculating the QAS threshold at step 3, the face images from the original noisy set that achieve a mean similarity to the identity reference set R above threshold T_{QAS} are considered as inliers. Therefore, the final identity clean set C is defined as following:

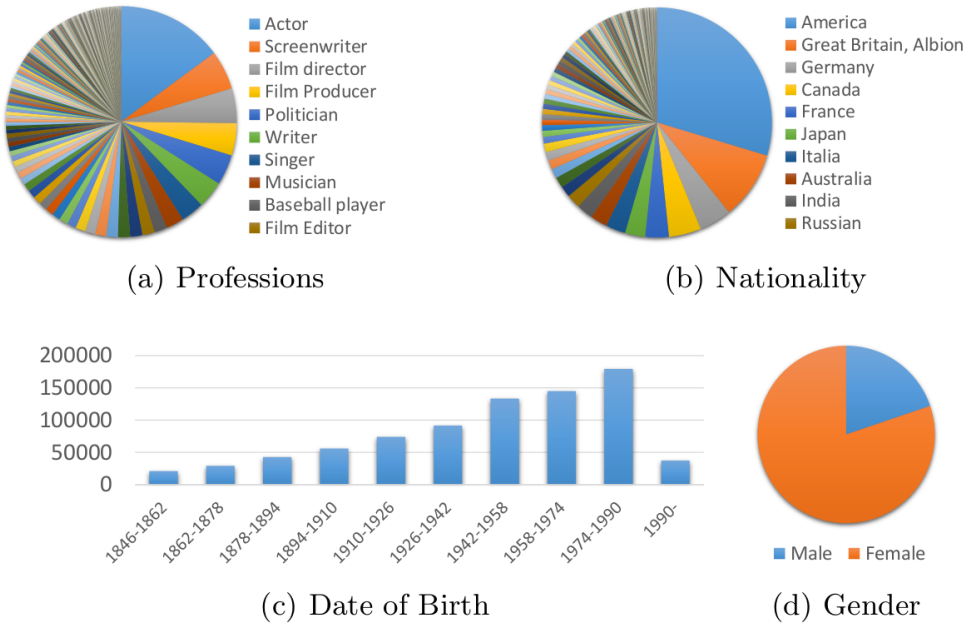
$$C = \{I \mid \text{sim}(I, R) > T_{\text{QAS}}), I \in S\} \tag{3.4}$$

where S is the noisy identity set, $\text{sim}(\cdot)$ is the function that calculates the mean similarity between the face image I and the face images in the set R using cosine similarity measure, T_{QAS} is QAS similarity threshold, varying with the face qualities.

By repeating the method for each identity faces set, we obtain the final cleaned face dataset.

Chapter 4

Evaluation Versus Semantic Bootstrapping Cleaning



(a) Professions (b) Nationality

(c) Date of Birth (d) Gender

Figure 4.1: Properties of the MS-Celeb-1M Database.[1]

4.1 MS-Celeb-1M.v1 Database Description

MS-Celeb-1M.v1 database [20, 1] contains 100K subjects and about 10 million images. This is a subset of the one million celebrity list collected by the authors from a knowledge graph called freebase. The authors used public search engines to collect approximately 100 images for each celebrity, resulting in about 10M web images. The one million celebrity list includes people with more than 2000 different professions and come from more than 200 distinct countries/regions. It also covers all major ethnic groups of the world and has a large age range. Some sample images from the dataset are shown in Figure 1.

As mentioned in the paper [1], the authors did not manually remove noise in the data set because the size of the data set is beyond the scale of manual labeling. In some cases the percent of outliers (face with wrong id label) is more than 70 percent of the total faces in an identity faces set. The outliers for one identity can contain faces of 10 different identities. The authors in [1] left the problem of id label cleaning open. We would like to take up the challenge by applying our id labels cleaning method on this dataset.

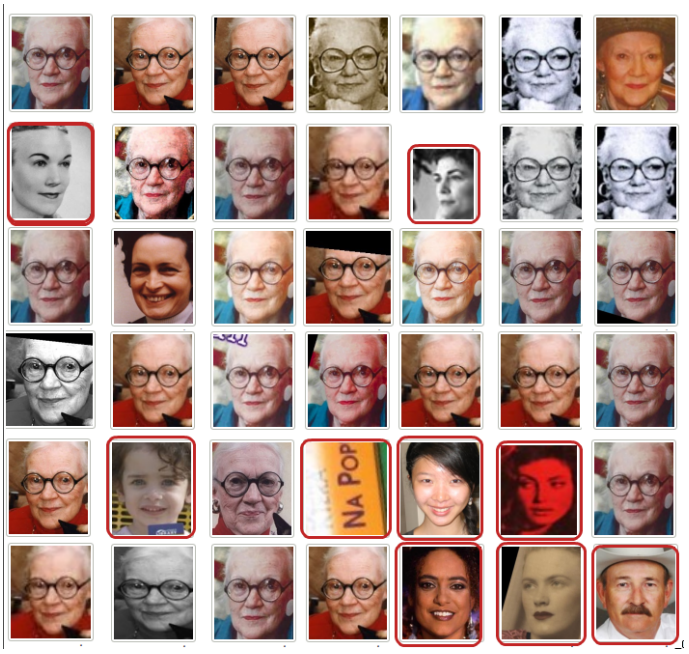


Figure 4.2: Sample images of a subject in the MS-Celeb-1M database. Noise images are highlighted in red boxes.

4.2 Experiment Settings

4.2.1 Face detection and alignment settings

All the face images are detected, aligned, converted to grayscale images and normalized into a size of 144×144 for the training data, and 140×140 for the testing data. We use the Openface[21, 22] library to detect facial landmarks. The mouth, ears, and eyes from detected landmarks are used in the face normalization and alignment process.

4.2.2 Similarity measure settings

Since our id label cleaning method uses face similarity measure, we train a lightCNN on CASIA-WebFace dataset using the same settings as in [2]. Table 5.1 shows the deep network architecture used to extract the features. . The momentum is set to 0.9, the weight decay is set to $5e-4$ and the learning rate is set to $1e-3$. The fully connected layer "eltwise_fc1" which has 256 dimensions is used to extract deep features. The similarity measure is based on the cosine similarity competition.

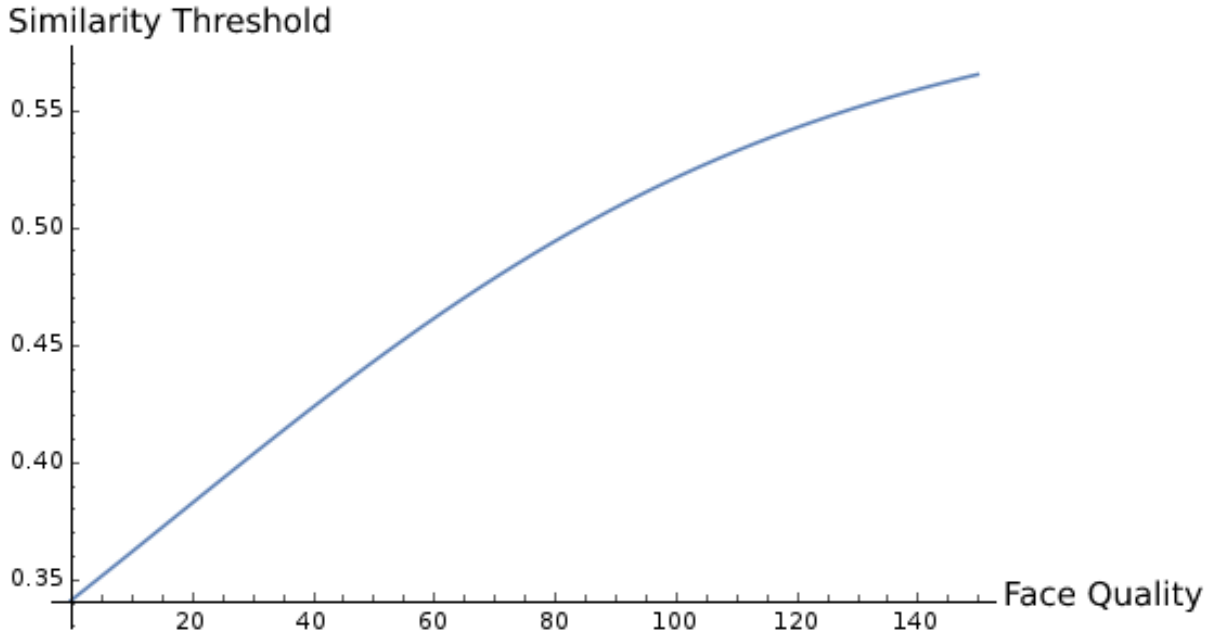


Figure 4.3: The quality adaptive similarity threshold function $T_{\text{QAS}}(\cdot)$ which used in the MS-Celeb-1M.v1 id label cleaning experiment. The similarity threshold values increase as the quality of the face images increase.

4.2.3 MS-Celeb-1M.v1 dataset identity label cleaning settings

Before starting the actual id label cleaning process, we need to estimate the values of the cleaning parameters which are appropriate to clean MS-Celeb-1M.v1, e.g., the mean quality score ($\overline{Q}$), the initial reference similarity threshold (T_{init}) and both the minimum and maximum thresholds in the quality-based similarity function (T_{min}, T_{max}). The mean quality score ($\overline{Q}$), is the mean quality score of all the faces in the dataset and is equal to 54. In order to find the best values for the other parameters, we defined a validation set of 40 identities. Our validation experiments show that the best clean result is obtained when $T_{\text{init}} = 0.25, T_{\text{min}} = 0.34, T_{\text{max}} = 0.63$. Figure 4.3 shows the function $T_{\text{QAS}}(\cdot)$ with the mentioned settings.

4.3 MS-Celeb-1M.v1 ID Label Cleaning Results

Our final quality-based cleaned 'MS-Celeb-1M.v1' contains $88,176$ identities and $4,517,039$ face images. The average number of images per identity is 49. From here on, we denote the cleaned version of 'MS-Celeb-1M.v1' as 'MS-Celeb-1M-Clean' dataset. Compared to the se-

	# of subjects	# of images
MS-Celeb-1M	100K	approx. 10M
MS-1M-1R [2]	79,077	4,086,798
MS-1M-2R [2]	79,077	5,049,824
Our MS-Celeb-1M-Clean	88,176	4,517,039

Table 4.1: Comparison between our MS-Celeb-1M-Clean dataset and Semantic bootstrapping clean datasets

mantic bootstrapping id label cleaning results sets (MS-1M-1R and MS-1M-2R), our method is able to keep around $10K$ more identities. Figure 4.4 shows the comparison between the number of identities of our MS-Celeb-1M-Clean dataset and the semantic bootstrapping dataset MS-1M-2R. Our MS-Celeb-1M-Clean dataset has 10,961 identities that are not exist in MS-1M-2R. However, $1,862$ identities are in MS-1M-2R could be falsely removed from ours, which means our method could be improved further to include more identities. Since we limited our method to use LightCNN deep architecture and Webface dataset in purpose of comparison with the semantic bootstrapping, using better models and more data to generate the face image features could overcome what it looks like some limitation.
Additionally, to show the effectiveness of our method on correctly classifying the low-quality images, we visually compare to the semantic bootstrapping method. Figure 4.6 shows examples of identities mainly with low-quality face images, are correctly kept by our method but are falsely considered as noise by the semantic bootstrapping method. We see there are a number of low-quality faces with blurry, low resolution, large pose change and partially covered faces in these examples. Also, Figure 4.5 shows the distribution of faces number per identity in MS-Celeb-1M-Clean.

4.3.1 Low-to-high quality face verification comparison

To compare our method to semantic bootstrpping [2] , we use MS-Celeb-1M-Clean to train the deep network proposed in [2] using the same training settings. Using the same settings but our cleaned data to do the training helps making a fair comparison between our and semantic bootstrapping versions of MS-Celeb-1M.v1 face dataset. Since the main goal of our method is preserving low-quality face images, we designed two low-to-high quality

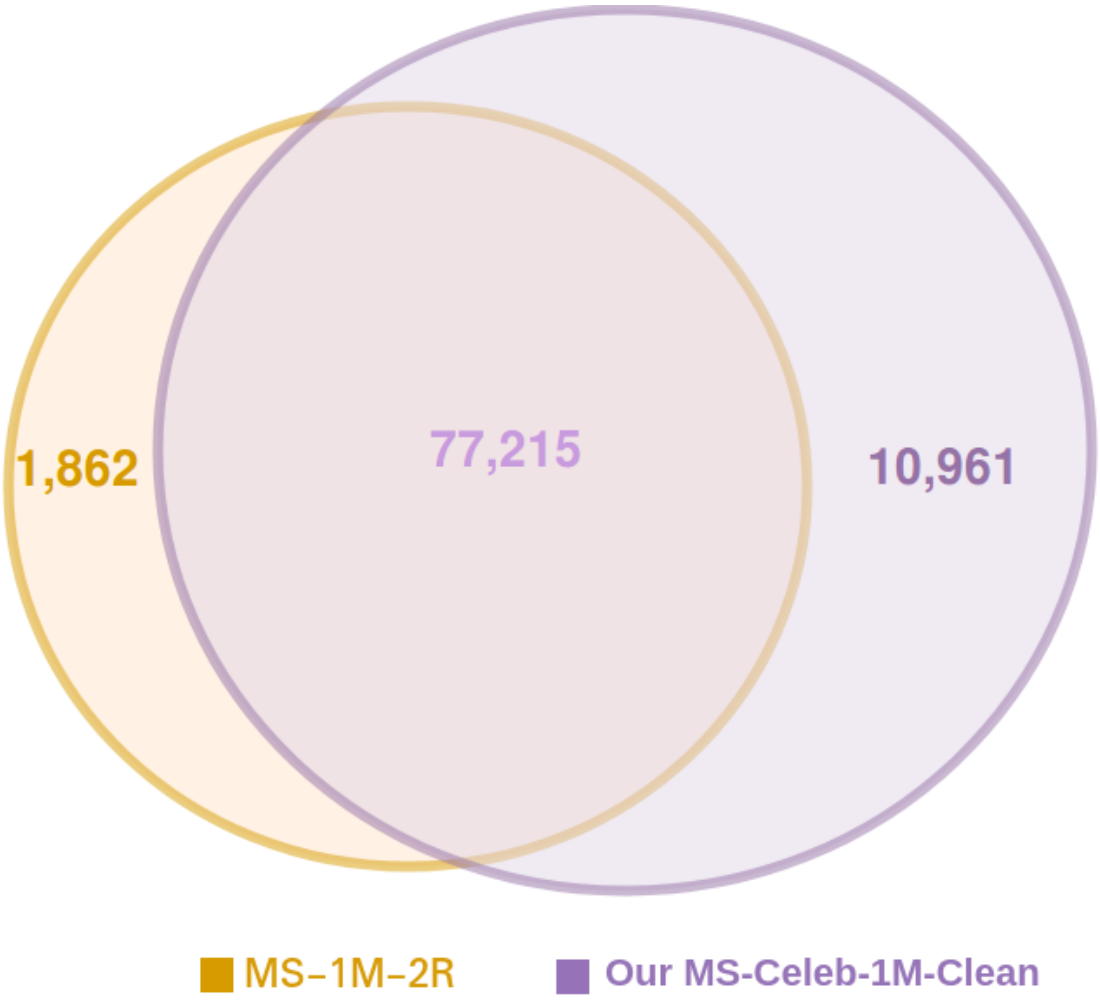


Figure 4.4: Our MS-Celeb-1M-Clean and MS-M1-2R (semantic bootstrapping) identity sets of the MS-Celeb-1M id label cleaning results (77,215 overlapped identities, 10,961 identities are only in our MS-Celeb-1M-Clean and 1,862 identities are only in MS-1M-2R).

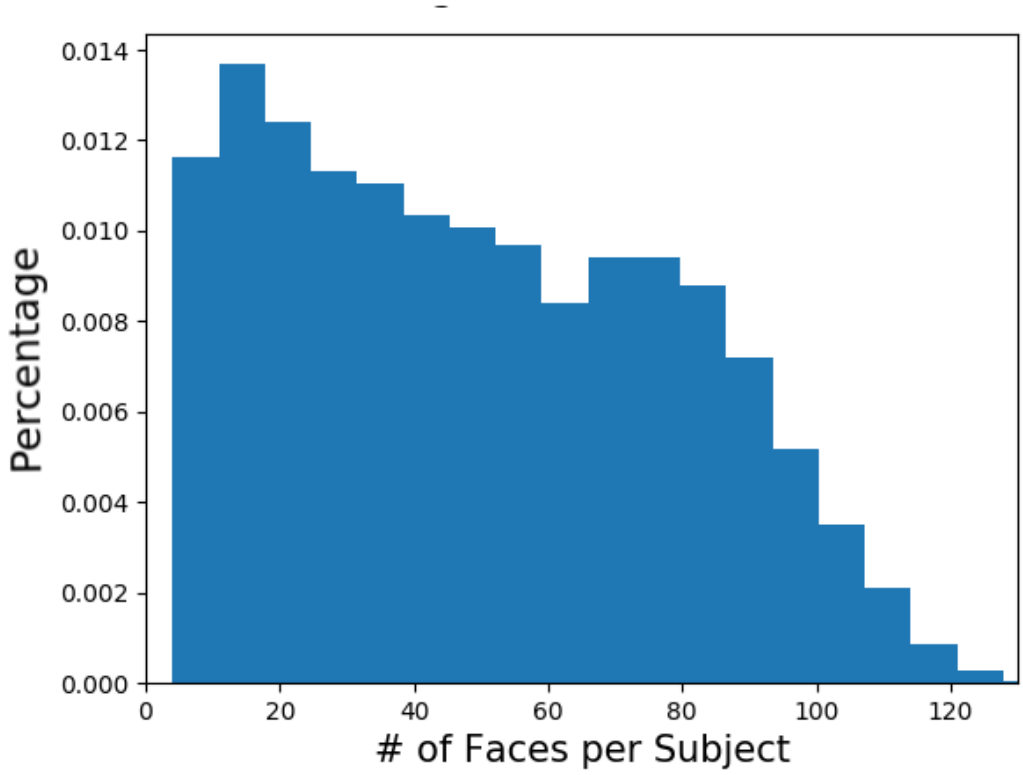


Figure 4.5: Distribution of the number of the images per subject for our MS-Celeb-1M-Clean dataset.

Figure 4.6: Some examples from our MS-Celeb-1M-Clean dataset which were falsely classified by MS-M1-2R (semantic bootstrapping) as noise.

face verification experiments using two different face datasets. In the following sections, we present the performance comparisons between our face model and the semantic bootstrapping model on the IJB-A [23] and FaceScrub [3] low-to-high quality face verification experiments.

IJB-A Experiment: Using th protocol in [24], IJB-A dataset is divided into two subsets based on the face image quality: 1) $10,089$ high-quality images and 2) 362 low-quality images. To perform the low-to-high quality experiments, we choose $6,676$ positive pairs and $3,645,542$ negative pairs. Each pair contains one low and one high-quality images. Our model is able to obtain 6% higher Verification rate (VR) at false acceptance rate (FAR) equal to 10^{-3} than the bootstrapping model, where our model achieves 50% $VR@FAR = 10^{-3}$, while the bootstrapping model achieves 44% $VR@FAR = 10^{-3}$. The ROC curve comparison is shown in Figure 4.7. Additionally, the verification accuracy comparison is given in Table 4.2. based on the results, our model achieves better face verification rates for various false acceptance rates on the IJB-A dataset, compared to the bootstrapping model.

FaceScrub Experiment: Similarly and using the protocol in [24], the FaceScrub dataset is divided into two subsets based on the quality: 1) $1,543$ high-quality images, and 2) $6,196$ low-quality images. To perform the low-to-high quality experiments, we generated $18,978$ positive pairs and $9,541,450$ negative pairs. Each pair contains one low and one high-quality image. Similar to the recognition results on IJB-A dataset, our model is able to obtain 6% higher Verification rate (VR) than the bootstrapping model at false acceptance rate (FAR) equal to 10^{-3}, where our model achieves 48% $VR@FAR = 10^{-3}$, while the bootstrapping model [2] achieves 42% $VR@FAR = 10^{-3}$. The ROC curve performance comparison is shown in Figure 4.8. Additionally, the verification accuracy comparisons are given in Table 4.3. Again, our model outperforms the bootstrapping model by achieving better face verification rates for various false acceptance rates on the FaceScrub dataset.

Based on the results of IJB-A and FaceScrub low-to-high-quality face verification experiments, we can say that compared to [2], our clean version of MS-Celeb-1M.v1 contains more face variations in terms of the face quality and better training data for low-to-high face matching tasks.

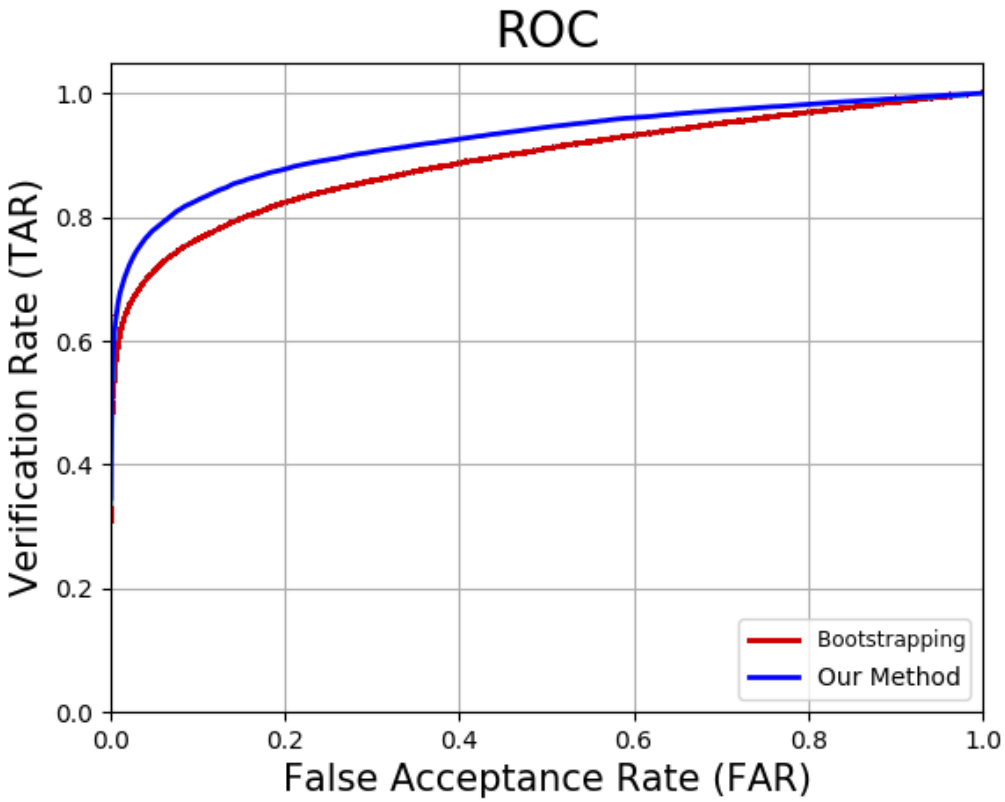


Figure 4.7: ROC comparison on IJB-A low-to-high quality face verification experiments.

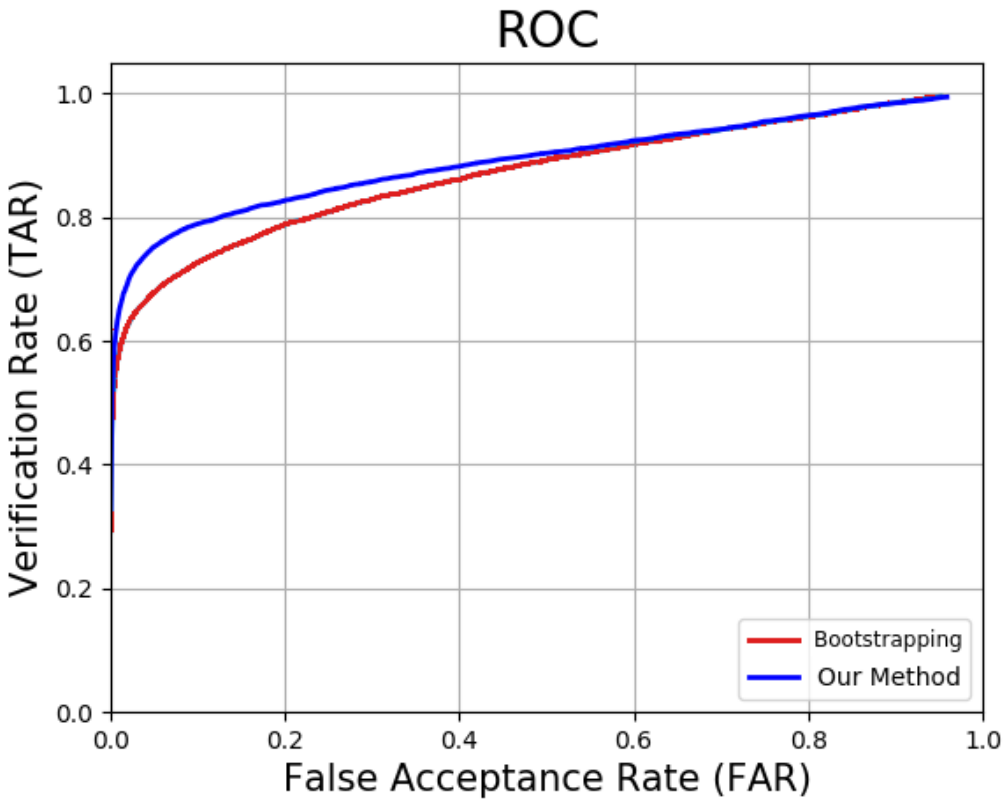


Figure 4.8: ROC comparison on FaceScrub low-to-high quality face verification experiments.

	Bootstrapping [2]	**Our model**
$FAR = 0.001$	0.44	**0.50**
$FAR = 0.01$	0.60	**0.67**
$FAR = 0.1$	0.76	**0.82**
EER	0.18	**0.14**
AUC	0.88	**0.92**

Table 4.2: Performance Comparison on IJB-A low-to-high quality face verification experiments

	Bootstrapping [2]	**Our model**
$FAR = 0.001$	0.42	**0.48**
$FAR = 0.01$	0.58	**0.64**
$FAR = 0.1$	0.72	**0.78**
EER	0.20	**0.18**
AUC	0.87	**0.89**

Table 4.3: Performance Comparison on FaceScrub low-to-high quality face verification experiments

Chapter 5

Comparison Versus Human Annotation

Up to now, human labeling is still considered as the best possible annotation and cleaning method even though it is time consuming and error prone to some extent. For this reason, we decided to evaluate our cleaning method using a manually cleaned dataset. Then we compare the result with another state-of-the-art cleaning method proposed by Ng and Winkler's [3] that has also been compared to a manually labeled dataset.

5.1 CACD Dataset

CACD [25] is a large dataset collected for cross-age face recognition in 2014, which includes $2,000$ identities of $162,815$ face images. As indicted in their paper the dataset might contain noise because they could accidentally collect images of other celebrities in the same event or movie. The fact that it is noisy, makes it good candidate to test our cleaning method. We manually cleaned a subset of the CACD dataset for our experiment. We chose 500 random identity of $40,757$ face images and manually annotated the faces as inliers or outliers. Our manual cleaning found out $6,967$ outliers in this chosen subset of the CACD.

5.2 Experiment Settings

5.2.1 Identity label cleaning settings

To clean those 500 identities with our proposed method, we used the same settings as we used to clean MS-Celeb-1M.v1, except we set the maximum and minimum similarity threshold to higher values, where $T_{\text{min}} = 0.36$ and $T_{\text{max}} = 0.66$. Higher similarity threshold gives better cleaning results, since there are more high-quality faces overall in the CACD dataset compared to MS-Celeb-1M.v1.

5.2.2 Face detection and alignment settings

Similar to our experiments in the previous chapter, all the face images are detected, aligned, converted to grayscale images and normalized into a size of 144×144 for the training data, and 140×140 for the testing data. We use the Openface[21, 22] library to detect facial landmarks. The mouth, ears, and eyes from detected landmarks are used in the face normalization and alignment process.

5.2.3 Similarity measure settings

Similar to our experiments in the previous chapter, lightCNN model is trained on CASIA-WebFace dataset using the same settings as in [2] to calculate the face features in order to measure faces similarity. Table 5.1 shows the lightCNN architecture. The momentum is set to 0.9, the weight decay is set to $5e-4$ and the learning rate is set to $1e-3$. The fully connected layer "eltwise_fc1" which has 256 dimensions is used to extract deep features. The similarity measure is based on the cosine similarity competition.

Type	Filter Size /Stride, Pad	Output Size	#Params
Conv1	$5 \times 5/1, 2$	$128 \times 128 \times 96$	2.4K
MFM1	-	$128 \times 128 \times 48$	-
Pool1	$2 \times 2/2$	$64 \times 64 \times 48$	-
Conv2_x	$\begin{bmatrix} 3 \times 3/1, 1 \\ 3 \times 3/1, 1 \end{bmatrix} \times 1$	$64 \times 64 \times 48$	82K
Conv2a	$1 \times 1/1$	$64 \times 64 \times 96$	4.6K
MFM2a	-	$64 \times 64 \times 48$	-
Conv2	$3 \times 3/1, 1$	$64 \times 64 \times 192$	165K
MFM2	-	$64 \times 64 \times 96$	-
Pool2	$2 \times 2/2$	$32 \times 32 \times 96$	-
Conv3_x	$\begin{bmatrix} 3 \times 3/1, 1 \\ 3 \times 3/1, 1 \end{bmatrix} \times 2$	$32 \times 32 \times 96$	662K
Conv3a	$1 \times 1/1$	$32 \times 32 \times 192$	18K
MFM3a	-	$32 \times 32 \times 96$	-
Conv3	$3 \times 3/1, 1$	$32 \times 32 \times 384$	331K
MFM3	-	$32 \times 32 \times 192$	-
Pool3	$2 \times 2/2$	$16 \times 16 \times 192$	-
Conv4_x	$\begin{bmatrix} 3 \times 3/1, 1 \\ 3 \times 3/1, 1 \end{bmatrix} \times 3$	$16 \times 16 \times 192$	3981K
Conv4a	$1 \times 1/1$	$16 \times 16 \times 384$	73K
MFM4a	-	$16 \times 16 \times 192$	-
Conv4	$3 \times 3/1, 1$	$16 \times 16 \times 256$	442K
MFM4	-	$16 \times 16 \times 128$	-
Conv5_x	$\begin{bmatrix} 3 \times 3/1, 1 \\ 3 \times 3/1, 1 \end{bmatrix} \times 4$	$16 \times 16 \times 128$	2356K
Conv5a	$1 \times 1/1$	$16 \times 16 \times 256$	32K
MFM5a	-	$16 \times 16 \times 128$	-
Conv5	$3 \times 3/1, 1$	$16 \times 16 \times 256$	294K
MFM5	-	$16 \times 16 \times 128$	-
Pool4	$2 \times 2/2$	$8 \times 8 \times 128$	-
fc1	-	512	4,194K
MFM_fc1	-	256	-
Total	-	-	12,637K

Table 5.1: The architectures of the Light CNN-29 model.

Dataset	# identities	# images	# outliers
Ng & Winkler[3]	20	5,791	794
Ours	500	40,757	6,967
Method	Recall	Precision	F1 Score
Ng & Winkler[3]	0.72	0.52	0.60
Ours	0.76	0.58	0.66

Table 5.2: The data size and id label cleaning performance results for the comparison with human annotation experiment.

5.3 CACD Cleaning Results and the comparison to Ng and Winkler's results

Note that, it was not possible for us to perform a direct comparison with Ng and Winkler's cleaning method [3], since their codes and datasets are not publicly available. So we perform an ad-hoc study by applying our cleaning method on a manually annotated noisy subset of CACD [25] dataset and measure the recall and precision. Then, we compare the precision-recall results with their published result. Our argument is that, even though we do not perform our comparative analysis using the same dataset, if we use a much larger dataset with much more noisy identity labels and still get a better precision-recall curve than them, then our cleaning method could be better than theirs.

Ng and Winkler [3] method identifies the identity label outliers by formulating the problem as a quadratic programming (QP) problem that combines the outputs of an outlier detection classifier and a gender classifier, enforcing visual similarity among the inliers, while at the same time constrains to at most one face per image to be an inlier.

Our method successfully detected 76% of the outliers (TP rate) but removed 11% of the inliers (FN rate). Comparing to Ng and Winkler's [3] method, our id label cleaning method outperforms their reported results in terms of both the recall and precision. Our id label cleaning results have a recall of 0.76 and precision of 0.58, whereas their method reported 0.72 recall and 0.52 precision. Note that, our test dataset is much larger compared to [3], their test set contains 5791 face images from 20 people, with 794 of them being outliers.

Compared to theirs, our test dataset has 25 times more identities of $40,757$ face images with $6,967$ outliers.

Chapter 6

Conclusion And Future Works

6.1 Future Works

In the future, this work can be applied to other face based labels like age and gender beside the identity labels which is proposed in this work. There are many face datasets with noisy age labels collected from internet and have many low-quality faces, and developing quality-based age labels cleaning method is important to generate better datasets for training better age estimation models.
Additionally, face quality control is important part of our method. Exploring other face quality predicting methods could improve our results. Also, other deep models structure could be explored beside the lightCNN as alternate for our face similarity measurement module.
Our cleaned version of MS-Celeb-1M.v1 has $10,000$ more identities than the best current cleaned version of it, and contains more low-quality faces. Therefore, releasing MS-Celeb-1M-Clean dataset can be useful for other researchers in the face identification area.

6.2 Conclusion

Cleaning large-scale face datasets has become a major challenge recently. We have presented a novel method for cleaning very large-scale face image datasets using a face image quality assessment scheme. Our method has shown that it can more efficiently solve the identity label noise problem in a large face dataset. Our high-to-low-quality face verification experiments on Facescrub and IJB-A datasets have shown the effectiveness of our method in face data cleaning by keeping more low-quality face images. Our cleaned version of MS-Celeb-1M.v1 has $10,000$ more identities than the bootstrapping based cleaned version [2], and contains more low-quality faces. Our method not only has generated better training dataset for low-quality face verification, but also produced higher recall and precision than a previous method, when compared to a human annotations on a subset of CACD datset.

References

[1] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao, *MS-Celeb-1M: A Dataset and Benchmark for Large-Scale Face Recognition*, pp. 87–102. Cham: Springer International Publishing, 2016.

[2] X. Wu, R. He, Z. Sun, and T. Tan, "A Light CNN for Deep Face Representation with Noisy Labels," *ArXiv e-prints*, Nov. 2015.

[3] H. W. Ng and S. Winkler, "A data-driven approach to cleaning large face datasets," in *2014 IEEE International Conference on Image Processing (ICIP)*, pp. 343–347, Oct 2014.

[4] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *British Machine Vision Conference*, vol. 1, p. 6, 2015.

[5] X. Zhang, L. Zhang, X. J. Wang, and H. Y. Shum, "Finding celebrities in billions of web images," *IEEE Transactions on Multimedia*, vol. 14, pp. 995–1007, Aug 2012.

[6] A. Bansal, A. Nanduri, C. Castillo, R. Ranjan, and R. Chellappa, "UMDFaces: An Annotated Face Dataset for Training Deep Networks," *ArXiv e-prints*, Nov. 2016.

[7] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," *CoRR*, vol. abs/1411.7923, 2014.

[8] Y. Sun, X. Wang, and X. Tang, "Hybrid deep learning for face verification," in *2013 IEEE International Conference on Computer Vision*, pp. 1489–1496, Dec 2013.

[9] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," tech. rep., Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[10] L. Wolf, T. Hassner, and I. Maoz, "Face recognition in unconstrained videos with matched background similarity," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 529–534, June 2011.

[11] K. Anantharajah, S. Denman, S. Sridharan, C. Fookes, and D. Tjondronegoro, "Quality based frame selection for video face recognition," in *2012 6th International Conference on Signal Processing and Communication Systems*, pp. 1–5, Dec 2012.

[12] K. Anantharajah, S. Denman, D. Tjondronegoro, S. Sridharan, C. Fookes, and X. Guo, "Quality based frame selection for face clustering in news video," in *2013 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pp. 1–8, Nov 2013.

[13] M. De Marsico, M. Nappi, and D. Riccio, "Measuring measures for face sample quality," in *Proceedings of the 3rd International ACM Workshop on Multimedia in Forensics and Intelligence*, MiFor '11, (New York, NY, USA), pp. 7–12, ACM, 2011.

[14] A. Abaza, M. A. Harrison, T. Bourlai, and A. Ross, "Design and evaluation of photometric image quality measures for effective face recognition," *IET Biometrics*, vol. 3, no. 4, pp. 314–324, 2014.

[15] K. Nasrollahi, T. B. Moeslund, and M. Rahmati, "Summarization of surveillance video sequences using face quality assessment," *International Journal of Image and Graphics*, vol. 11, no. 02, pp. 207–233, 2011.

[16] H. Sellahewa and S. A. Jassim, "Image-quality-based adaptive face recognition," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, pp. 805–813, April 2010.

[17] Y. Wong, S. Chen, S. Mau, C. Sanderson, and B. C. Lovell, "Patch-based probabilistic image quality assessment for face selection and improved video-based face recognition," in *CVPR 2011 WORKSHOPS*, pp. 74–81, June 2011.

[18] J. Chen, Y. Deng, G. Bai, and G. Su, "Face image quality assessment based on learning to rank," *IEEE Signal Processing Letters*, vol. 22, pp. 90–94, Jan 2015.

[19] W. J. Scheirer, P. J. Flynn, C. Ding, G. Guo, V. Struc, M. A. Jazaery, K. Grm, S. Dobrisek, D. Tao, Y. Zhu, J. Brogan, S. Banerjee, A. Bharati, and B. RichardWebster, "Report on the btas 2016 video person recognition evaluation," in *2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pp. 1–8, Sept 2016.

[20] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao, "MS-Celeb-1M: A Dataset and Benchmark for Large-Scale Face Recognition," *ArXiv e-prints*, July 2016.

[21] T. Baltrušaitis, P. Robinson, and L. P. Morency, "Openface: An open source facial behavior analysis toolkit," in *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1–10, March 2016.

[22] T. Baltrusaitis, P. Robinson, and L. P. Morency, "Constrained local neural fields for robust facial landmark detection in the wild," in *2013 IEEE International Conference on Computer Vision Workshops*, pp. 354–361, Dec 2013.

[23] B. F. Klare, B. Klein, E. Taborsky, A. Blanton, J. Cheney, K. Allen, P. Grother, A. Mah, M. Burge, and A. K. Jain, "Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1931–1939, June 2015.

[24] G. Guo and N. Zhang, "What is the challenge for deep learning in unconstrained face recognition?," in *2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018)*, pp. 436–442, May 2018.

[25] B.-C. Chen, C.-S. Chen, and W. H. Hsu, *Cross-Age Reference Coding for Age-Invariant Face Recognition and Retrieval*, pp. 768–783. Cham: Springer International Publishing, 2014.